# Explications sur le simulateur – Version universitaire détaillée (octobre 2025)

Cette version du document explicatif du simulateur intègre les mises à jour conceptuelles et fonctionnelles effectuées entre septembre et octobre 2025. Elle présente les fondements théoriques du modèle, les variables clés, ainsi que les options avancées liées à la récompense, à la dynamique des mèmes, et à la limitation probabiliste des événements.

## 1. Introduction générale

Le simulateur modélise la diffusion et la transformation des opinions au sein d’une population d’agents interconnectés. Chaque agent possède une opinion, une prévalence (ou prégnance cognitive), et un degré d’influence. Les interactions sont structurées sous forme de réseaux dynamiques où les opinions évoluent par imitation, résistance, et renforcement de groupe. Ce modèle s’inscrit dans la lignée des approches en systèmes complexes adaptatifs (Holland, 1995 ; Epstein & Axtell, 1996) et s’inspire des travaux sur la dynamique d’opinion (Deffuant et al., 2010 & 2011) ainsi que sur la cognition distribuée et la propagation des représentations (Sperber, 1996 ; Boyd & Richerson, 2005).

Le modèle introduit plusieurs dimensions cognitives et sociales : l’effet de groupe, la force mémique des représentations, le rôle des méta-influenceurs et des agents ordinaires, ainsi que des mécanismes de récompense simulant la consolidation du charisme ou du leadership social. Les événements exogènes peuvent également perturber ou polariser le système, en modulant directement les distributions d’opinion et de prévalence.

## 2. Dynamique de base

Les agents évoluent sur un continuum d’opinions variant de -1 (pôle négatif) à +1 (pôle positif). Leur prévalence correspond au degré de représentation interne des arguments qui soutiennent cette opinion. Chaque itération (‘tick’) simule une interaction aléatoire entre agents reliés : l’un agit comme émetteur, l’autre comme récepteur. La probabilité d’adoption dépend de plusieurs facteurs : la différence de prévalence, le degré d’influence, la proximité d’opinion, et l’alignement du groupe social. Les connexions inter-agents sont continuellement actualisées pour refléter les affinités ou les divergences croissantes.

## 3. Système de récompense

Le système de récompense introduit une dimension dynamique de renforcement social. Lorsqu’un agent réussit à influencer un autre agent, il reçoit un bonus d’influence (tx-bonus). Ce bonus augmente la probabilité d’influencer à nouveau lors des interactions suivantes, simulant ainsi la consolidation d’une réputation, d’un prestige ou d’un capital symbolique. Le paramètre reward-step contrôle la vitesse d’accumulation du bonus, tandis que reward-cap en fixe la limite supérieure. La variable reward-decay permet d’introduire une décroissance progressive de cette récompense, traduisant l’oubli social ou la perte d’influence dans le temps.

Exemple d’utilisation : un chercheur peut simuler l’émergence de leaders d’opinion en fixant reward-step à 0.1 et reward-cap à 0.5, ce qui crée des agents capables d’accumuler une influence supérieure après plusieurs succès consécutifs.

## 4. Dynamique des mèmes

L’activation de l’option use-memes? transforme la dynamique de l’opinion en un processus cognitif fondé sur la propagation de mèmes. Chaque agent possède un stock de mèmes favorables (meme-plus) et défavorables (meme-minus). La somme de ces deux valeurs détermine la force de la prévalence, tandis que leur différence indique la direction de l’opinion. Lors d’une interaction, les mèmes peuvent être transmis selon la polarité de l’émetteur : les agents alignés tendent à renforcer leurs mèmes cohérents, tandis que les opposés subissent une ‘fuite’ contrôlée par meme-anti-leak.

Exemple : si meme-anti-leak = 0.2, un agent perd une partie de ses mèmes contraires lorsqu’il reçoit un argument cohérent avec son opinion. Ce mécanisme permet d’explorer la formation des bulles cognitives ou des chambres d’écho.

## 5. Méta-influenceurs et veto (metablock)

Les méta-influenceurs (agents jaunes) représentent des acteurs d’influence majeure – médias dominants, institutions, leaders politiques, ou figures symboliques. Lorsqu’ils sont soumis à la contrainte metablock = true, ces agents ne peuvent pas changer de polarité d’opinion. Ils peuvent néanmoins renforcer leur position actuelle. Cette règle empêche les renversements idéologiques brusques et maintient une structure d’influence stable, souvent observée dans les systèmes sociaux hiérarchisés.

Exemple : dans une simulation politique, metablock=true permet de reproduire la stabilité idéologique des partis dominants face à des fluctuations d’opinion dans la population.

## 6. Limitation de la proportion d’agents affectés par un événement

Le paramètre event-prob-max (0.00–1.00 ; défaut : 1.00) permet de définir la proportion maximale d’agents susceptibles d’être touchés par un événement externe. Lors du déclenchement d’un événement (bouton event ou auto\_event), chaque agent tire une valeur aléatoire u ∈ [0,1]. Si u ≤ event-prob-max, l’agent est affecté selon la logique de l’événement : changement d’opinion (event\_size), variation de prévalence (prev\_change), et respect des bornes low\_meme–high\_meme et low-prev–high-prev.

• 1.00 (100 %) : tous les agents éligibles participent à l’événement – choc global (ex. : crise systémique, pandémie, révolution médiatique).  
• 0.10 (10 %) : choc partiel – effet localisé d’une campagne d’opinion ou d’une rumeur virale.  
• 0.01 (1 %) : micro-perturbations locales – discussions interpersonnelles ou influence diffuse.

## Répétition des événements (repeat-event & event-pace)

Cette section formalise la capacité de programmation d’événements exogènes répétés. Elle complète la logique existante d’auto\_event et d’event-prob-max.

* Paramètres et widgets :
* repeat-event (switch) — lorsqu’il est activé, chaque événement planifié se reprogramme automatiquement.
* event-pace (slider numérique ≥ 1 tick) — pas entre deux occurrences consécutives de l’événement répété.
* tick-event (input) — tick auquel le prochain événement est attendu.
* auto\_event (switch) — déclencheur automatique à l’itération tick-event (conserver l’usage du bouton event pour un tir manuel).
* Logique opérationnelle (conforme au code fourni) :
* • Si auto\_event = ON et que iter = tick-event : l’événement est exécuté.
* • Si repeat-event = ON : après exécution, tick-event ← tick-event + max(1, arrondi(event-pace)).
* • Si repeat-event = OFF : aucune reprogrammation automatique n’est faite ; l’utilisateur peut ajuster tick-event manuellement (ou réactiver plus tard).
* • Si auto\_event = OFF : à chaque tick, le modèle positionne tick-event ← iter + event-pace ; lorsque vous repassez auto\_event = ON, le prochain événement surviendra ≈ event-pace ticks plus tard.
* Rappel : l’intensité de l’événement est modulée par event-prob-max (0–1). À chaque occurrence, chaque agent i est affecté si U(0,1) ≤ event-prob-max ; sinon il est ignoré pour cette occurrence.
* Bonnes pratiques :
* • Utiliser une valeur entière pour event-pace (p. ex. 25, 50, 100) ; en cas de valeur non entière, l’arrondi est appliqué et un pas minimal de 1 tick est garanti.
* • Éviter event-pace = 0, qui empêcherait toute reprogrammation (le code protège néanmoins en imposant un minimum de 1).
* • Lors d’un changement de event-pace en cours de simulation, la nouvelle période s’applique à partir de la prochaine reprogrammation (après la prochaine occurrence).
* Exemples d’usage réalistes :

1. 1) Chocs médiatiques périodiques : repeat-event = ON, event-pace = 100, event-prob-max = 0.30. Produit une impulsion toutes les 100 itérations qui touche ~30 % des agents éligibles.
2. 2) Campagne courte et intense : repeat-event = ON, event-pace = 10, event-prob-max = 0.60, prev\_change > 0. Accélère la montée en prévalence et peut déstabiliser la polarisation locale.
3. 3) Choc unique de calibration : auto\_event = ON, repeat-event = OFF, tick-event = 2, event-prob-max = 1.00. Déclenche un seul événement global au tout début, utile pour tester la sensibilité du modèle.
4. 4) Impacts diffus : repeat-event = ON, event-pace = 50, event-prob-max = 0.05. Simule des micro-perturbations régulières affectant une petite fraction d’agents.

* Cas limites / Remarques :
* • Si tick-event < iter au moment où vous activez auto\_event, aucun événement ne se produira tant que tick-event ne sera pas repositionné (par exemple via la logique auto\_event OFF → ON, ou en ajustant manuellement tick-event).
* • Avec metablock = ON, les méta-influenceurs touchés par un événement ne peuvent pas inverser leur signe d’opinion ; ils peuvent toutefois se renforcer (magnitude).
* • Si use-memes? = ON, l’événement peut être suivi d’une réinitialisation cohérente du stock de mèmes (selon votre code), ce qui impacte la recomposition de prevalence et d’opinion.

## 7. Applications et interprétation

Ce simulateur permet d’explorer la co-évolution des opinions et des réseaux sociaux à travers une approche multi-niveaux : interactions individuelles, effets de groupe, et contraintes structurelles. Il peut être utilisé dans divers contextes : analyse de polarisation politique, diffusion d’innovations, dynamique de désinformation ou apprentissage collectif.

L’ajustement des paramètres tels que group-impact-weight, reward-step, meme-anti-leak ou event-prob-max offre un outil heuristique permettant de tester des hypothèses issues des sciences cognitives, de la psychologie sociale et de la sociologie numérique.

## Références

1. Boyd, Stephen P & Richerson, Peter J. (2005) *The origin and evolution of cultures,* Oxford University Press
2. Epstein, Joshua M & Axtell, Robert L (2008) *Growing Artificial Societies: Social Science from the Bottum Up,* MIT Press
3. Deffuant, Guillaume, Cioffi-Revilla, Claudio, Takadama, Keiki (2010) *Simulating Interacting Agents and Social Phenomena,* Springer
4. Deffuant, Guillaume & Nigel, Gilbert (2011) *Viability and Resilience of Complex Systems: Concepts, Methods and Case Studies from Ecology and Society (Understanding Complex Systems),* Springer
5. Holland, John H. (1995) *Hidden order : how adaptation builds complexity*, Perseus Books
6. Sperber, Dan (1995) *La Contagion des idées,* Odile Jacob